



БЕЗОПАСНОСТЬ БАЗ ДАННЫХ

ФИО преподавателя: Селин А.А., канд. техн. наук

КОНЦЕПЦИИ ЗАЩИТЫ ДАННЫХ В СИСТЕМАХ БАЗ ДАННЫХ

Учебные вопросы:

1. Принципы безопасного ведения данных в СБД
2. SQL-инъекция
3. Роли и привилегии коллективной обработки данных

ПРИНЦИПЫ БЕЗОПАСНОГО ВЕДЕНИЯ ДАННЫХ В СБД

Аудит в базах данных

Аудит в базах данных — системный процесс получения объективных качественных и количественных оценок о текущем состоянии информационной безопасности системы баз данных в соответствии с определёнными критериями и показателями.

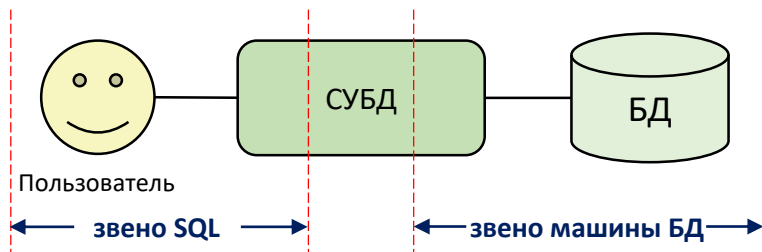
Основные функции аудита баз данных:

- оценка легитимности доступа – проверка и оценка процедур разграничения данных и тщательного контроля доступа;
- регистрация транзакций – ведение журнала регистрации транзакций, как критических операций в отношении целостности данных;
- регистрация действий пользователей – журнализация запросов пользователей, не связанных с транзакциями (выборка, переиндексация данных, агрегирование, копирование, представление,...);
- расследование инцидентов нарушения информационной безопасности.

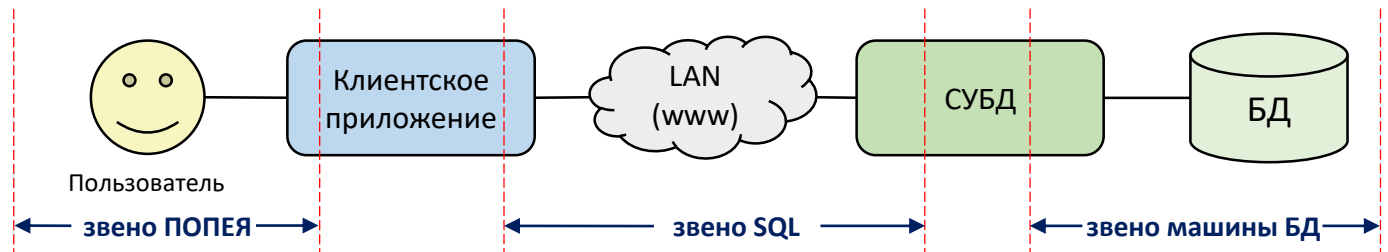


1. До 40% увеличение нагрузки на администраторов.
2. Отсутствует возможность контроля деятельности администраторов.
3. Не возможен контроль действий пользователей в трехзвенных и n-звенных архитектурах.

Двухзвенная архитектура обработки данных



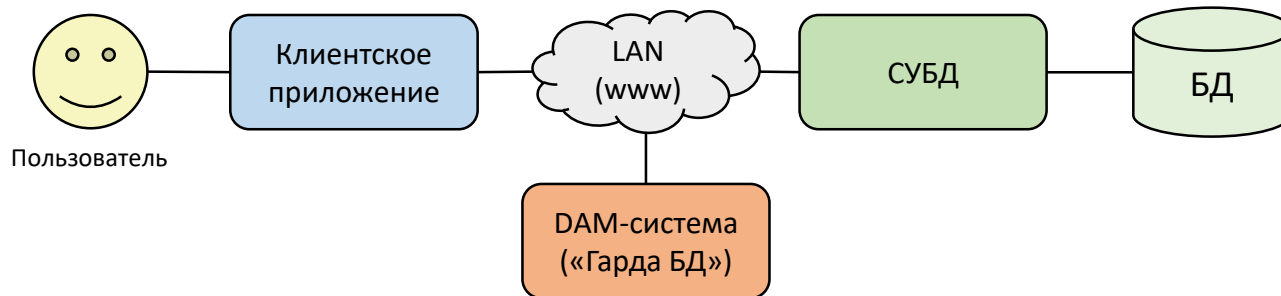
Трёхзвенная архитектура обработки данных



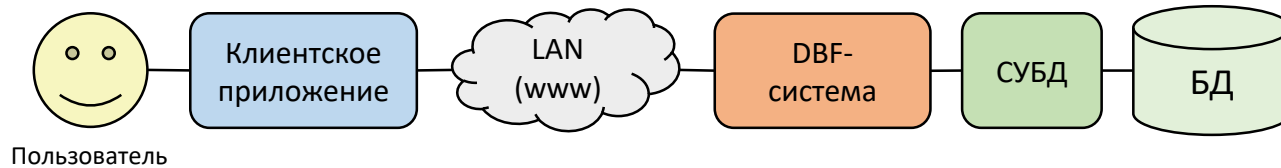
ПРИНЦИПЫ БЕЗОПАСНОГО ВЕДЕНИЯ ДАННЫХ В СБД

Автоматизированные системы защиты баз данных

Database Activity Monitoring – DAM



Database Firewall – DBF



Основные функции систем защиты БД:

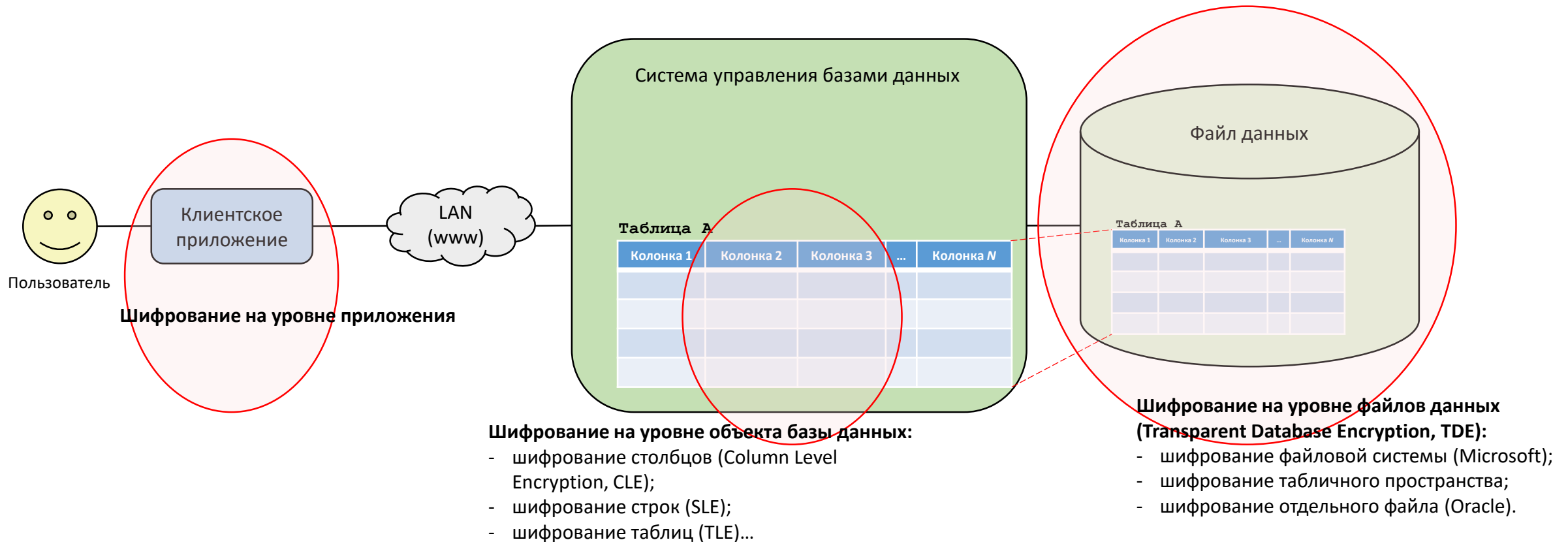
- классификация данных – определения местонахождения критичной для предприятия с точки зрения информационной безопасности информации;
- проверка соответствия настроек СУБД экстремумным показателям информационной безопасности;
- восстановление матрицы безопасности (поиск «мертвых» учетных записей, расширенных привилегий доступа, неиспользуемых предоставленных полномочий,...);
- гибкое формирование отчетности по информационной безопасности, в том числе корреляция с интеллектуальными системами информационной безопасности.



1. Увеличение телетрафика в телекоммуникационной подсистеме, особенно, для DBF-систем.
2. Возможность восстановления структуры базы данных по перехваченным запросам и отчетам АС защиты базы данных.
3. Отсутствует возможность управления целостностью хранимых данных.

ПРИНЦИПЫ БЕЗОПАСНОГО ВЕДЕНИЯ ДАННЫХ В СБД

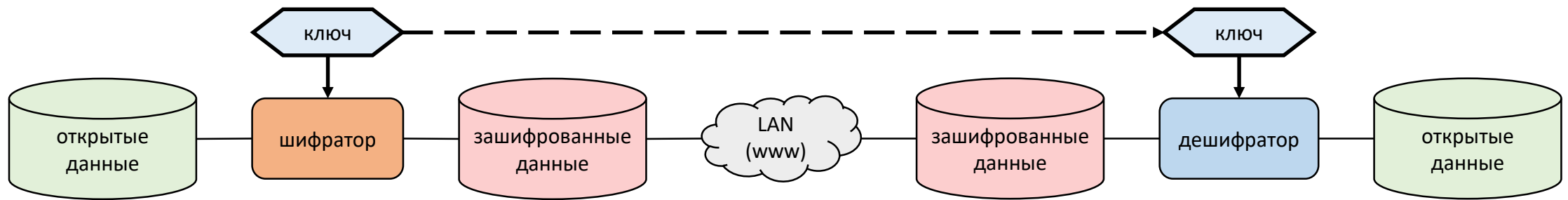
Шифрование данных



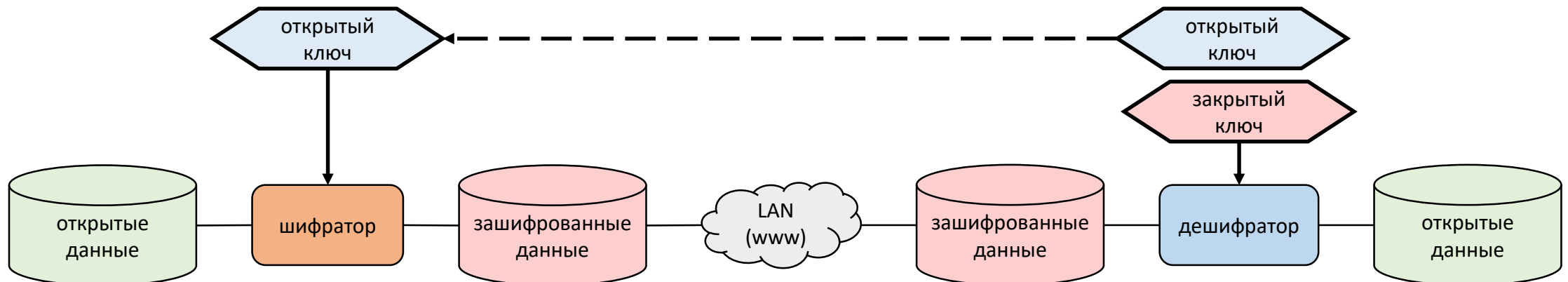
ПРИНЦИПЫ БЕЗОПАСНОГО ВЕДЕНИЯ ДАННЫХ В СБД

Шифрование данных

Шифрование с симметричным ключом



Шифрование с асимметричным ключом



ПРИНЦИПЫ БЕЗОПАСНОГО ВЕДЕНИЯ ДАННЫХ В СБД

Контроль и обеспечение целостности данных

Свойства информации, как объекта информационной безопасности:

доступность (availability) [средства обеспечения: представление, витрина данных];
целостность (integrity) [способы обеспечения: отказоустойчивость, аварийное восстановление];
конфиденциальность (confidentiality).

Constraints – свойства ограничения целостности для одной таблицы:

- C1. Дублирование строк таблицы не допускается.
- C2. Порядок строк в отношении не существенен.
- K1. Каждое отношение имеет, по крайней мере, один ключ.
- K2. Значение ключа уникально идентифицирует кортеж отношения.
- K3. Никакое подмножество атрибутов ключа не обладает свойством уникальности идентификации.
- K4. Первичный ключ не допускается обновлять или оставлять без значения.
- K5. Связи между отношениями должны поддерживаться только с помощью ключей.
- A1. Атрибуты отношений должны быть определены по типу и формату представления.
- A2. Диапазон области допустимых значений атрибута может быть ограничен только средствами языка запросов.
- A3. При присвоении каждому столбцу уникального имени роли порядок столбцов не существенен.

Referential Integrity – правила ссылочной целостности для соединенных таблиц:

RESTRICT – правило не удаления;

CASCADE – каскадное изменение;

SET DEFAULT – установка в значение по умолчанию;

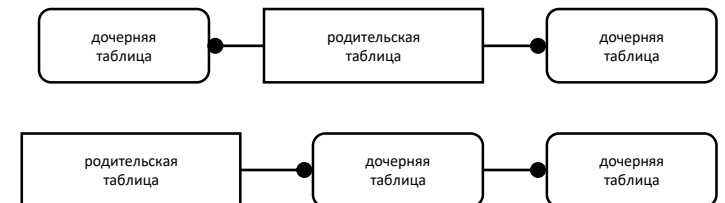
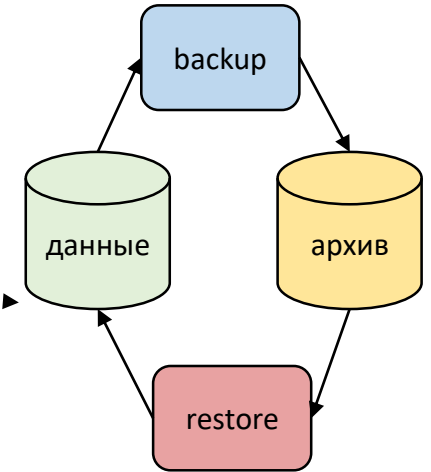
SET NULL – неопределенная ссылка;

NONE – правило не реагирования.

Connection Trap – ловушки соединения, механизм обеспечения целостности для трех соединенных таблиц:

branch trap (ловушка разветвления)

gap trap (ловушка разрыва)



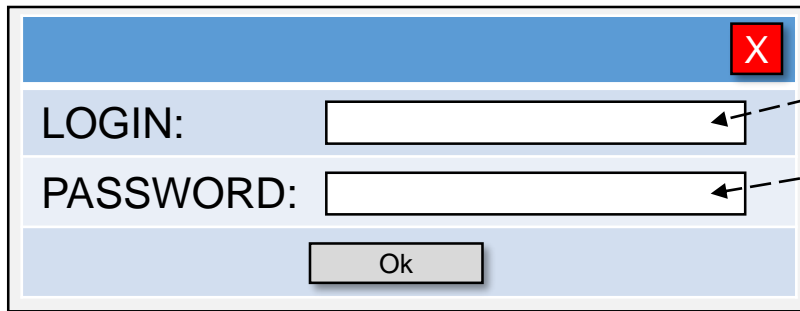
SQL-ИНЪЕКЦИЯ

SQL-инъекция – это способ нападения на базу данных с применением легальных методов, основанных на реляционной природе языка структурированных запросов, встроенного в СУБД. Используются принципы вычисления предикатов первого порядка, на которых основано реляционное исчисление.

SQL-инъекция основывается на динамическом языке запросов – только при необходимости формировать текст SQL-предложения непосредственно в ходе выполнения кода прикладной программы.

А. Нелегитимное подключение к базы данных

```
SELECT ... FROM dba_users WHERE user_name = '<имя учетной записи>' AND password = '_____';
```



A screenshot of a login dialog box. It has a blue header bar with a red 'X' button in the top right corner. Below the header, there are two input fields: 'LOGIN:' and 'PASSWORD:'. At the bottom, there is an 'Ok' button. Dashed arrows point from the input fields to the SQL query above and below, indicating where the injected payload is placed.

aaa' OR 1 = 1'

```
SELECT ... WHERE user_name = '<имя учетной записи>' AND password = 'aaa' OR 1 = 1' ';
```

ИСТИНА

AND

ЛОЖЬ

OR

ИСТИНА

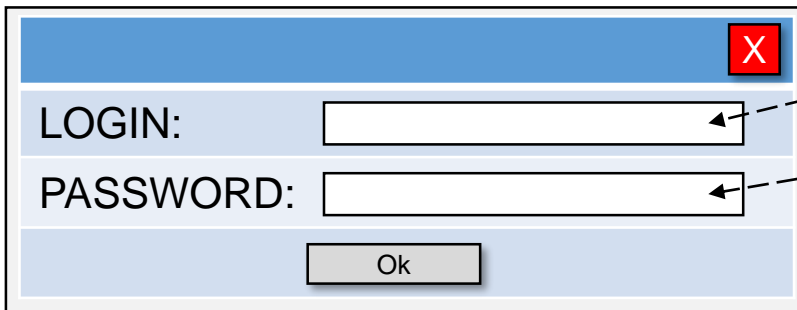
=

ИСТИНА

SQL-ИНЪЕКЦИЯ

А. Нелегитимное подключение к базы данных

```
SELECT ... FROM dba_users WHERE user_name = ' ' AND password = '<любая_строка>';
```



A login dialog box with a blue header bar containing a red 'X' button. Below the header are two input fields: 'LOGIN:' and 'PASSWORD:'. At the bottom is an 'Ok' button.

```
a@a.com' OR 1 = 1 LIMIT 1; --
```

```
SELECT ... WHERE user_name = 'a@a.com' OR 1 = 1 LIMIT 1; -- AND password = 'любая_строка';
```

Исполняемая часть запроса

Отсеченная часть запроса

SQL-ИНЪЕКЦИЯ

Б. Добавление нелегальной инструкции в SQL-предложение

Имеется форма справочника для получения, например, номера телефона сотрудника учреждения, в которой следует указать его имя.

```
SELECT phone FROM employee WHERE emp_name = '<ввод_имени_сотрудника>';
```

Вводим любое имя и добавляем конструкцию вида ` UNION SELECT username FROM all_users `.

```
SELECT phone FROM employee WHERE emp_name = 'Петров В.В.'  
UNION  
SELECT username FROM all_users '';
```

Выдача будет содержать номер телефона Петрова В.В. (если такой зарегистрирован в базе данных) и список имен всех учетных записей БД.

При тех же исходных данных добавляем конструкцию вида ` OR EXIST (SELECT 1 FROM sysdual) `.

```
SELECT phone FROM employee WHERE emp_name = 'Петров В.В.'  
OR EXIST (SELECT 1 FROM sysdual) '';
```

Выдача будет содержать все зарегистрированные в БД номера телефонов.

В. Подмена встроенной функции хакерским SQL-кодом

Г. Навязывание predetermined ошибки выполнения интерпретатора SQL

SQL-ИНЪЕКЦИЯ

Правила защиты от атак с использованием SQL-инъекций:

Ввод пользовательских данных не должен быть доверенным. Вводимые параметры учетной записи должны быть санированы (выполнить парсинг введенных данных с целью выявления подозрительных сигнатур), прежде чем они станут аргументами функций с динамически формируемыми SQL-предложениями.

Хранимые процедуры, которые инкапсулируют **SQL-предложения** и обрабатывают все входные данные только в качестве собственных параметров.

Применение статического SQL-предложения, где передаваемые пользователем параметры используются только в качестве значений переменных в SQL-предложениях, созданных разработчиком, и не могут повлиять на синтаксис и семантику выполняемого запроса.

Регулярные выражения обнаружения потенциально вредоносного кода (фрагментов SQL-предложений, передаваемых в качестве параметров, особенно в формах ввода данных пользователем) и его удаления перед выполнением **SQL-предложений**.

Применение принципа минимальных полномочий для всех, без исключения, учетных записей. Это ограничит перечень действий, которые можно инициировать от имени взломанной учетной записи.

Сообщения об ошибках не должны содержать конфиденциальную информацию. Для уведомления пользователя об ошибке необходимо использовать общие фразы. Всегда следует заменять типовые сообщения СУБД об ошибках.

РОЛИ И ПРИВИЛЕГИИ КОЛЛЕКТИВНОЙ ОБРАБОТКИ ДАННЫХ

Привилегия – право пользователя (пользовательского процесса) на «прикосновение» к данным с указанием, что он может с ними сделать (вставить, удалить, модифицировать, выбрать,..) или выполнение какой-либо работы в базе данных, в том числе подключение к ней.

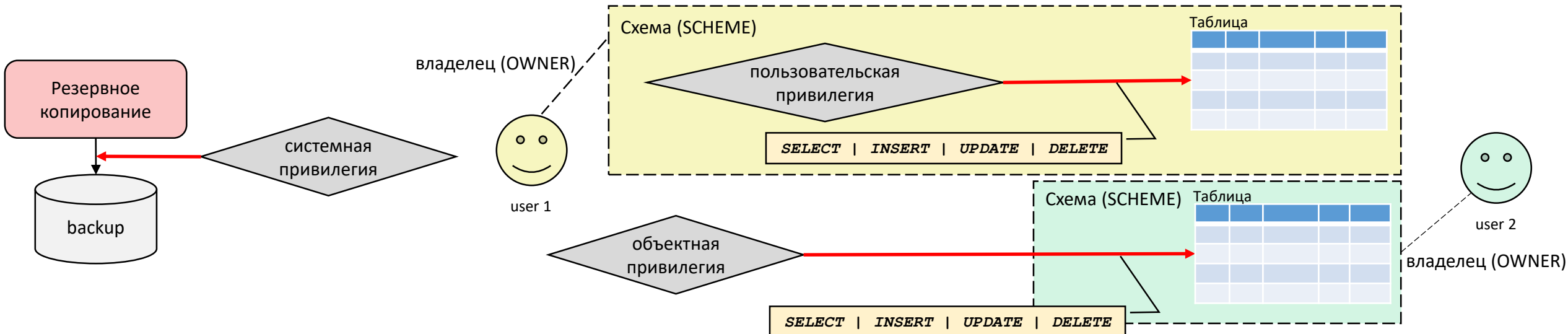
GRANT <имя_привилегии> TO <имя_учетной_записи/имя_роли>;

REVOKE <имя_привилегии> FROM <имя_учетной_записи/имя_роли>;

Привилегии: **системные** – право пользователя (пользовательского процесса) на применение системной функции;

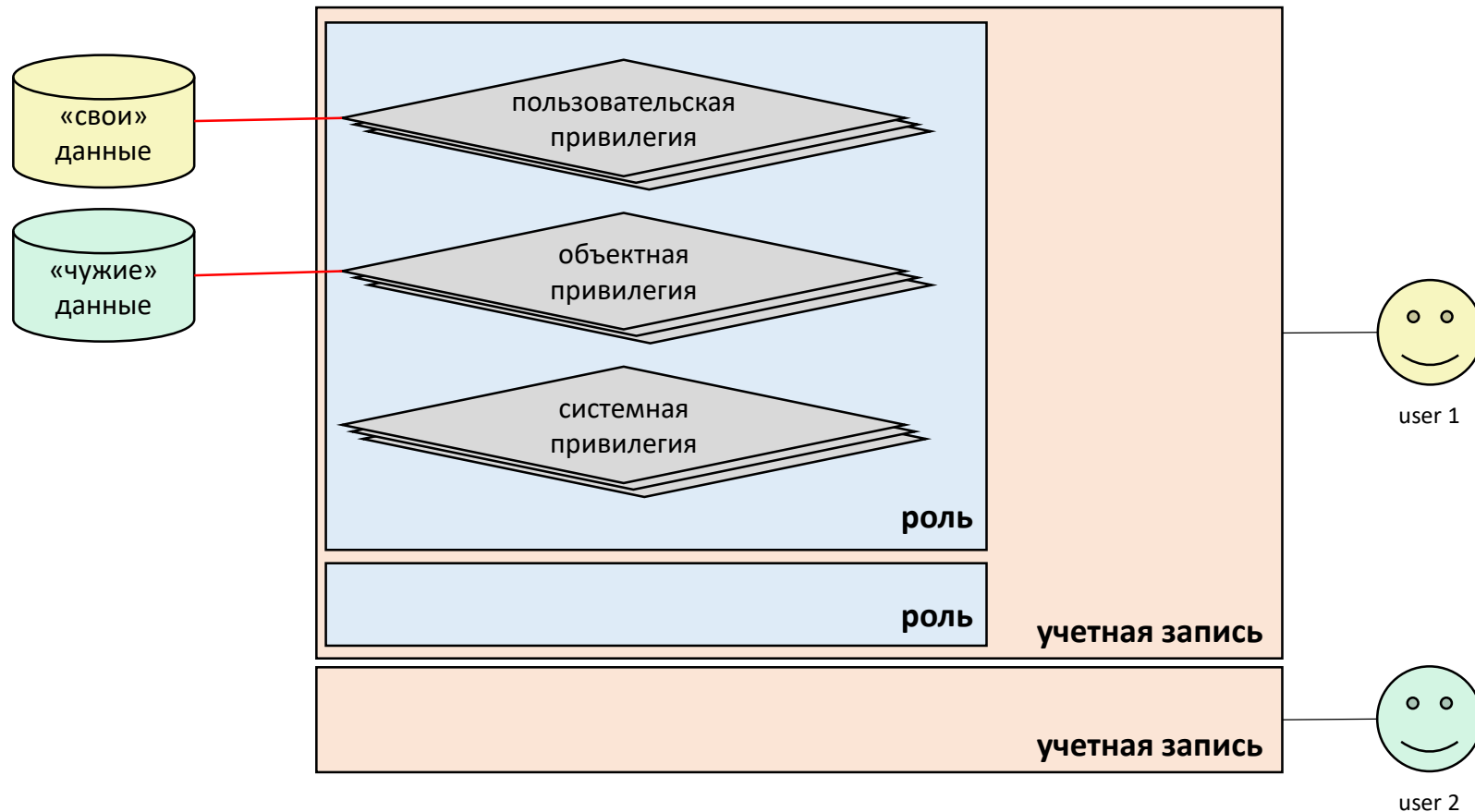
пользовательские – право пользователя по манипулированию данными и пользовательскими объектами «собственной» схемы;

объектные – право пользователя на манипулирование данными и пользовательскими объектами «чужой» схемы.



РОЛИ И ПРИВИЛЕГИИ КОЛЛЕКТИВНОЙ ОБРАБОТКИ ДАННЫХ

Роль – комплект привилегий для типовой работы пользователя в соответствии с его функциональными обязанностями в учреждении (например, кассир билетной кассы, бухгалтер по проводке оплаты труда, оператор склада, специалист по снабжению, руководитель основного подразделения, менеджер торгового зала,...). Комплект привилегий зависит от бизнес-модели (совокупности бизнес-процессов), реализованной в учреждении.



РОЛИ И ПРИВИЛЕГИИ КОЛЛЕКТИВНОЙ ОБРАБОТКИ ДАННЫХ

Учетная запись – совокупность ролей (в том числе и единственная роль) для работы пользователя с данными и объектами базы данных (в том числе и «чужими» – принадлежащими схеме другого пользователя).

```
CREATE USER <имя_учетной_записи>
  [WITH [SUPERUSER]
        [CREATEDB]
        [CREATEROLE]
        [INHERIT]
        [LOGIN]
        [REPLICATION]
        [BYPASSRLS]
        [CONNECTION LIMIT <число_подключений>]
        [PASSWORD <пароль>]
        [VALID UNTIL <дата_время>]
        [IN ROLE <имя_роли>]
        [IN GROUP <имя_группы>]
        [ROLE <список_ролей>]
        [ADMIN <список_ролей>]
        [USER <список_ролей>]
        [SYSID <uid>]];
```

- наделение правами администратора базы данных
- наделение правом создавать экземпляр базы данных
- наделение правом создания новых ролей
- наследование привилегий роли, к которой принадлежит учетная запись
- наделение правом подключения к базе данных
- разрешение на выполнение репликации узла базы данных
- наделение правом игнорирования политики безопасности данных
- ограничение числа одновременных подключений к базе данных
- задание значения пароля
- установка момента блокировки учетной записи
- задание ролей учетной записи
- задание принадлежности к группе учетных записей
- задание списка дочерних ролей (членов группы)
- создание группы администраторов
- устаревшая спецификация привилегии ROLE
- обеспечение совместимости с системами, использующими uid

Литература:

1. **Смирнов, С. Н.** Безопасность систем баз данных [Текст]: учеб. пособие для вузов по специальностям в области информационной безопасности. — М.: Гелиос АРВ, 2007. — 350 с.
2. **Федин, Ф. О.** Информационная безопасность баз данных. Ч. 1 [Электронный ресурс]: учебное пособие / Ф. О. Федин, О. В. Трубиенко, С. В. Чискидов. — М.: РТУ МИРЭА, 2020. — Электрон. опт. диск (ISO)
3. **Терьо, М.** Oracle. Руководство по безопасности [Текст] / М. Терьо, А. Ньюмен; Пер. с англ.. — М.: Лори, 2004. — 560 с.: ил.
4. **Советов, Б. Я.** Базы данных: теория и практика : Учебник для вузов / Б. Я. Советов, В. В. Цехановский, В. Д. Чертовской. — М.: Высш. шк., 2005. — 464 с.: ил.
5. **Саймон, А.** Безопасность баз данных. // СУБД № 1, 1997 г. — с. 78 — 95.
6. **Кузнецов, С. Д.** Основы баз данных: курс лекций : учеб. пособие для студентов вузов, обучающихся по специальностям в обл. ин-форм. технологий / С. Д. Кузнецов. — Москва: Интернет-ун-т ин-форм. технологий, 2005. — 488 с.
7. **Смирнов, С. Н., Задворьев, И. С.** Работаем с ORACLE.: Учебное пособие/2-е изд., испр. и доп. — М: Гелиос АРВ, 2002 г. — 496 с.
8. **Кульба, В.В.** и др. Теоретические основы проектирования оптимальных структур распределенных баз данных. — М: СИНТЕГ, 1999 г. — 660 с.
9. Материалы сервера ORACLE/RE. www.oracle.ru/press/magazine/main.html
10. Материалы информационного ресурса WIKIPEDIA. https://ru.wikipedia.org/wiki/Разграничение_доступа_на_основе_атрибутов; <https://ru.wikipedia.org/wiki/Аутентификация>; https://ru.wikipedia.org/wiki/Многофакторная_аутентификация; https://ru.wikipedia.org/wiki/Сложность_пароля.